Value-at-Risk Optimization with Gaussian Processes

Quoc Phong Nguyen [1] **Zhongxiang Dai** [1] **Bryan Kian Hsiang Low** [1] **Patrick Jaillet** [2]

Abstract

Value-at-risk (VaR) is an established measure to assess risks in critical real-world applications with random environmental factors. This paper presents a novel VaR *upper confidence bound* (V-UCB) algorithm for maximizing the VaR of a black-box objective function with the first no-regret guarantee. To realize this, we first derive a confidence bound of VaR and then prove the existence of values of the environmental random variable (to be selected to achieve no regret) such that the confidence bound of VaR lies within that of the objective function evaluated at such values. Our V-UCB algorithm empirically demonstrates state-of-the-art performance in optimizing synthetic benchmark functions, a portfolio optimization problem, and a simulated robot task.

1. Introduction

Consider the problem of maximizing an expensive-to-compute black-box objective function f that depends on an *optimization variable* $\mathbf{x}$ and an *environmental random variable* $\mathbf{Z}$. Due to the randomness in $\mathbf{Z}$, the function evaluation $f(\mathbf{x}, \mathbf{Z})$ of f at $\mathbf{x}$ is a random variable. Though for such an objective function f, *Bayesian optimization* (BO) can be naturally applied to maximize its expectation $\mathbb{E}_{\mathbf{Z}}[f(\mathbf{x}, \mathbf{Z})]$ over $\mathbf{Z}$ (Toscano-Palmerin & Frazier, 2018), this maximization objective overlooks the *risks* of potentially undesirable function evaluations. These risks can arise from either (a) the realization of an unknown distribution of $\mathbf{Z}$ or (b) the realization of the random $\mathbf{Z}$ given that the distribution of $f(\mathbf{x}, \mathbf{Z})$ can be estimated well or that of $\mathbf{Z}$ is known. The issue (a) has been tackled by distributionally robust BO (Kirschner et al., 2020; Nguyen et al., 2020) which maximizes $\mathbb{E}_{\mathbf{Z}}[f(\mathbf{x}, \mathbf{Z})]$ under the worst-case realization of the distribution of $\mathbf{Z}$. To resolve the issue (b), the risk from the uncertainty in $\mathbf{Z}$ can be controlled via the mean-variance optimization framework (Iwazaki et al., 2020), *value-at-risk* (VaR), or *conditional value-at-risk* (CVaR) (Cakmak et al., 2020; Torossian et al., 2020). The work of Bogunovic et al. (2018) has considered *adversarially robust BO*, where $\mathbf{z}$ is controlled by an adversary deterministically.[1] In this case, the objective is to find $\mathbf{x}$ that maximizes the function under the worst-case realization of $\mathbf{z}$, i.e., $\operatorname{argmax}_{\mathbf{x}} \min_{\mathbf{z}} f(\mathbf{x}, \mathbf{z})$.

In this paper, we focus on case (b) where the distribution of $\mathbf{Z}$ is known (or well-estimated). For example, in agriculture, although farmers cannot control the temperature of an outdoor farm, its distribution can be estimated from historical data and controlled in an indoor environment for optimizing the plant yield. Given the distribution of $\mathbf{Z}$, the objective is to control the risk that the function evaluation $f(\mathbf{x}, \mathbf{z})$, for a $\mathbf{z}$ sampled from $\mathbf{Z}$, is small. One popular framework is to control the trade-off between the mean (viewed as reward) and the variance (viewed as risk) of the function evaluation with respect to $\mathbf{Z}$ (Iwazaki et al., 2020). However, quantifying the risk using variance implies indifference between positive and negative deviations from the mean, while people often have asymmetric risk attitudes (Goh et al., 2012). In our problem of maximizing the objective function, it is reasonable to assume that people are risk-averse towards only the negative deviations from the mean, i.e., the risk of getting low function evaluations. Thus, it is more appropriate to adopt risk measures with this asymmetric property, such as *value-at-risk* (VaR) which is a widely adopted risk measure in real-world applications (e.g., in banking (Basel Committee on Banking Supervision, 2006)). Intuitively, the risk that the random $f(\mathbf{x}, \mathbf{Z})$ is less than VaR at level $\alpha \in (0, 1)$ does not exceed α. Hence, by specifying a small value of α as 0.1, this risk is controlled to be at most 10%. Therefore, to maximize f while controlling the risk of undesirable function evaluations, we aim to maximize VaR of the random function $f(\mathbf{x}, \mathbf{Z})$ over $\mathbf{x}$.

The recent work of Cakmak et al. (2020) has used BO to maximize VaR and has achieved state-of-the-art empirical performances. They have assumed that we are able to select both $\mathbf{x}$ and $\mathbf{z}$ to query during BO, which is motivated by

[1]Department of Computer Science, National University of Singapore, Republic of Singapore [2]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, USA. Correspondence to: Quoc Phong Nguyen <qphong@comp.nus.edu.sg>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

[1]We use upper-case letter $\mathbf{Z}$ to denote the environmental random variable and lower-case letter $\mathbf{z}$ to denote its realization or a (non-random) variable.

fact that physical experiments can be studied by simulation (Williams et al., 2000). In the above agriculture example, we can control the temperature, lighting, and water ($\mathbf{z}$) in an indoor environment to search for the optimal amount of fertilizer ($\mathbf{x}$), which can then be used in an outdoor environment with random weather factors. The work of Cakmak et al. (2020) has exploited the ability to select $\mathbf{z}$ to model the function $f(\mathbf{x}, \mathbf{z})$ as a GP, which allows them to retain the appealing closed-form posterior belief of the objective function. To select the queries $\mathbf{x}$ and $\mathbf{z}$, they have designed a one-step lookahead approach based on the well-known *knowledge gradient* (KG) acquisition function (Scott et al., 2011). However, the one-step lookahead incurs an expensive nested optimization procedure, which is computationally expensive and hence requires approximations. Besides, the acquisition function can only be approximated using samples of the objective function f from the GP posterior and the environmental random variable $\mathbf{Z}$. While they have analysed the asymptotically unbiased and consistent estimator of the gradients, it is challenging to obtain a guarantee for the convergence of their algorithm. Another recent work (Torossian et al., 2020) has also applied BO to maximize VAR using an asymmetric Laplace likelihood function and variational approximation of the posterior belief. However, in contrast to Cakmak et al. (2020) and our work, they have focused on a different setting where the realizations of $\mathbf{Z}$ are not observed.

In this paper, we adopt the setting of Cakmak et al. (2020) which allows us to choose both $\mathbf{x}$ and $\mathbf{z}$ to query, and assume that the distribution of $\mathbf{Z}$ is known or well-estimated. Our contributions include:

Firstly, we propose a novel BO algorithm named *Value-at-risk Upper Confidence Bound* (V-UCB) in Section 3. Unlike the work of Cakmak et al. (2020), V-UCB is equipped with a no-regret convergence guarantee and is more computationally efficient. To guide its query selection and facilitate its proof of the no-regret guarantee, the classical GP-UCB algorithm (Srinivas et al., 2010) constructs a *confidence bound* of the objective function. Similarly, to maximize the VAR of a random function, we, for the first time to the best of our knowledge, construct a confidence bound of VAR (Lemma 2). The resulting confidence bound of VAR naturally gives rise to a strategy to select $\mathbf{x}$. However, it remains a major challenge to select $\mathbf{z}$ to preserve the no-regret convergence of GP-UCB. To this end, we firstly prove that our algorithm is no-regret as long as at the selected $\mathbf{z}$, the confidence bound of VAR *lies within* the confidence bound of the objective function. Next, we also prove that this query selection strategy is *feasible*, i.e., such values of $\mathbf{z}$, referred to as *lacing values* (LV), exist.

Secondly, although our theoretical no-regret property allows the selection of *any* LV, we design a heuristic to select an LV such that it improves our empirical performance over random selection of LV (Section 3.3). We also discuss the implications when $\mathbf{z}$ cannot be selected by BO and is instead randomly sampled by the environment during BO (Remark 1).

Thirdly, we show that adversarially robust BO (Bogunovic et al., 2018) can be cast as a special case of our V-UCB when the risk level α of VAR approaches 0 from the right and the domain of $\mathbf{z}$ is the support of $\mathbf{Z}$. In this case, adversarially robust BO (Bogunovic et al., 2018) selects the same input queries as those selected by V-UCB since the set of LV collapse into the set of minimizers of the lower bound of the objective function (Section 3.4).

Lastly, we provide practical techniques for implementing V-UCB with continuous random variable $\mathbf{Z}$ (Section 3.5): we (a) introduce *local neural surrogate optimization* with the *pinball loss* to optimize VAR, and (b) construct an objective function to search for an LV in the continuous support of $\mathbf{Z}$.

The performance of our proposed algorithm is empirically demonstrated in optimizing several synthetic benchmark functions, a portfolio optimization problem, and a simulated robot task in Section 4.

2. Problem Statement and Background

Let the objective function be defined as $f : \mathcal{D}_\mathbf{x} \times \mathcal{D}_\mathbf{z} \to \mathbb{R}$ where $\mathcal{D}_\mathbf{x} \subset \mathbb{R}^{d_x}$ and $\mathcal{D}_\mathbf{z} \subset \mathbb{R}^{d_z}$ are the bounded domain of the optimization variable $\mathbf{x}$ and the support of the environmental random variable $\mathbf{Z}$, respectively (d_x and d_z are the dimensions of $\mathbf{x}$ and $\mathbf{z}$, respectively). The support of $\mathbf{Z}$ is defined as the smallest closed subset $\mathcal{D}_\mathbf{z}$ of $\mathbb{R}^{d_z}$ such that $P(\mathbf{Z} \in \mathcal{D}_\mathbf{z}) = 1$. Let $\mathbf{z} \in \mathcal{D}_\mathbf{z}$ denote a realization of the random variable $\mathbf{Z}$. Let $f(\mathbf{x}, \mathbf{Z})$ denote a random variable whose randomness comes from $\mathbf{Z}$. The VAR of $f(\mathbf{x}, \mathbf{Z})$ at *risk level* $\alpha \in (0, 1)$ is defined as:

$$V_\alpha(f(\mathbf{x}, \mathbf{Z})) \triangleq \inf\{\omega : P(f(\mathbf{x}, \mathbf{Z}) \leq \omega) \geq \alpha\} \quad (1)$$

which implies the risk that $f(\mathbf{x}, \mathbf{Z})$ is less than its VAR at level α does not exceed α.

Our objective is to search for $\mathbf{x} \in \mathcal{D}_\mathbf{x}$ that maximizes $V_\alpha(f(\mathbf{x}, \mathbf{Z}))$ at a user-specified risk level $\alpha \in (0, 1)$. Intuitively, the goal is to find $\mathbf{x}$ where the evaluations of the objective function are as large as possible under most realizations of the environmental random variable $\mathbf{Z}$ (which is characterized by the probability of $1 - \alpha$).

The unknown objective function $f(\mathbf{x}, \mathbf{z})$ is modeled with a GP. That is, every finite subset of $\{f(\mathbf{x}, \mathbf{z})\}_{(\mathbf{x},\mathbf{z}) \in \mathcal{D}_\mathbf{x} \times \mathcal{D}_\mathbf{z}}$ follows a multivariate Gaussian distribution (Rasmussen & Williams, 2006). The GP is fully specified by its *prior* mean and covariance function $k_{(\mathbf{x},\mathbf{z}),(\mathbf{x}',\mathbf{z}')} \triangleq \text{cov}[f(\mathbf{x}, \mathbf{z}), f(\mathbf{x}', \mathbf{z}')]$ for all $\mathbf{x}, \mathbf{x}'$ in $\mathcal{D}_\mathbf{x}$ and $\mathbf{z}, \mathbf{z}'$ in $\mathcal{D}_\mathbf{z}$.

For notational simplicity (and w.l.o.g.), the former is assumed to be zero, while we use the *squared exponential* (SE) kernel as its bounded maximum information gain can be used for later analysis (Srinivas et al., 2010).

To identify the optimal $\mathbf{x}_* \triangleq \text{argmax}_{\mathbf{x}\in\mathcal{D}_\mathbf{x}} V_\alpha(f(\mathbf{x},\mathbf{Z}))$, BO algorithm selects an input query $(\mathbf{x}_t, \mathbf{z}_t)$ in the t-th iteration to obtain a noisy function evaluation $y_{(\mathbf{x}_t,\mathbf{z}_t)} \triangleq f(\mathbf{x}_t,\mathbf{z}_t) + \epsilon_t$ where $\epsilon_t \sim \mathcal{N}(0,\sigma_n^2)$ are i.i.d. Gaussian noise with variance σ_n^2. Given noisy observations $\mathbf{y}_{\mathcal{D}_t} \triangleq (y_{(\mathbf{x},\mathbf{z})})^\top_{(\mathbf{x},\mathbf{z})\in\mathcal{D}_t}$ at observed inputs $\mathcal{D}_t \triangleq \mathcal{D}_{t-1} \cup \{(\mathbf{x}_t,\mathbf{z}_t)\}$ ($\mathcal{D}_0$ is the initial observed inputs), the GP posterior belief of function evaluation at any input $(\mathbf{x},\mathbf{z})$ is a Gaussian $p(f(\mathbf{x},\mathbf{z})|\mathbf{y}_{\mathcal{D}_t}) \triangleq \mathcal{N}(f(\mathbf{x},\mathbf{z})|\mu_t(\mathbf{x},\mathbf{z}), \sigma_t^2(\mathbf{x},\mathbf{z}))$:

$$\begin{aligned}\mu_t(\mathbf{x},\mathbf{z}) &\triangleq \mathbf{K}_{(\mathbf{x},\mathbf{z}),\mathcal{D}_t}\mathbf{\Lambda}_{\mathcal{D}_t\mathcal{D}_t}\mathbf{y}_{\mathcal{D}_t}\ ,\\ \sigma_t^2(\mathbf{x},\mathbf{z}) &\triangleq k_{(\mathbf{x},\mathbf{z}),(\mathbf{x},\mathbf{z})} - \mathbf{K}_{(\mathbf{x},\mathbf{z}),\mathcal{D}_t}\mathbf{\Lambda}_{\mathcal{D}_t\mathcal{D}_t}\mathbf{K}_{\mathcal{D}_t,(\mathbf{x},\mathbf{z})}\end{aligned} \quad (2)$$

where $\mathbf{\Lambda}_{\mathcal{D}_t\mathcal{D}_t} \triangleq \left(\mathbf{K}_{\mathcal{D}_t\mathcal{D}_t} + \sigma_n^2\mathbf{I}\right)^{-1}$, $\mathbf{I}$ is the identity matrix of the same dimensions as $\mathbf{K}_{\mathcal{D}_t\mathcal{D}_t}$, $\mathbf{K}_{(\mathbf{x},\mathbf{z}),\mathcal{D}_t} \triangleq (k_{(\mathbf{x},\mathbf{z}),(\mathbf{x}',\mathbf{z}')})_{(\mathbf{x}',\mathbf{z}')\in\mathcal{D}_t}$, $\mathbf{K}_{\mathcal{D}_t,(\mathbf{x},\mathbf{z})} \triangleq \mathbf{K}^\top_{(\mathbf{x},\mathbf{z}),\mathcal{D}_t}$, and $\mathbf{K}_{\mathcal{D}_t\mathcal{D}_t} \triangleq (k_{(\mathbf{x}',\mathbf{z}'),(\mathbf{x}'',\mathbf{z}'')})_{(\mathbf{x}',\mathbf{z}'),(\mathbf{x}'',\mathbf{z}'')\in\mathcal{D}_t}$.

3. BO of VaR

Following the seminal work (Srinivas et al., 2010), we use the *cumulative regret* as the performance metric to quantify the performance of our BO algorithm. It is defined as $R_T \triangleq \sum_{t=1}^T r(\mathbf{x}_t)$ where $r(\mathbf{x}_t) \triangleq V_\alpha(f(\mathbf{x}_*,\mathbf{Z})) - V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))$ is the *instantaneous regret* and $\mathbf{x}_* \triangleq \text{argmax}_{\mathbf{x}\in\mathcal{D}_\mathbf{x}} V_\alpha(f(\mathbf{x},\mathbf{Z}))$. We would like to design a query selection strategy that incurs *no regret*, i.e., $\lim_{T\to\infty} R_T/T = 0$. Furthermore, we have that $\min_{t\le T} r(\mathbf{x}_t) \le R_T/T$, equivalently, $\max_{t\le T} V_\alpha(f(\mathbf{x}_t,\mathbf{Z})) \ge V_\alpha(f(\mathbf{x}_*,\mathbf{Z})) - R_T/T$. Thus, $\lim_{T\to\infty}\max_{t\le T} V_\alpha(f(\mathbf{x}_t,\mathbf{Z})) = V_\alpha(f(\mathbf{x}_*,\mathbf{Z}))$ for a no-regret algorithm.

The proof of the upper bound on the cumulative regret of GP-UCB is based on confidence bounds of the objective function (Srinivas et al., 2010). Similarly, in the next section, we start by constructing a confidence bound of $V_\alpha(f(\mathbf{x},\mathbf{Z}))$, which naturally leads to a query selection strategy for $\mathbf{x}_t$.

3.1. A Confidence Bound of $V_\alpha(f(\mathbf{x},\mathbf{Z}))$ and the Query Selection Strategy for $\mathbf{x}_t$

Firstly, we adopt a confidence bound of the function $f(\mathbf{x},\mathbf{z})$ from Chowdhury & Gopalan (2017), which assumes that f belongs to a *reproducing kernel Hilbert space* $\mathcal{F}_k(B)$ such that its RKHS norm is bounded $\|f\|_k \le B$.

Lemma 1 (Chowdhury & Gopalan (2017)**).** Pick $\delta \in (0,1)$ and set $\beta_t = (B + \sigma_n\sqrt{2(\gamma_{t-1} + 1 + \log 1/\delta)})^2$. Then, $f(\mathbf{x},\mathbf{z}) \in I_{t-1}[f(\mathbf{x},\mathbf{z})] \triangleq [l_{t-1}(\mathbf{x},\mathbf{z}), u_{t-1}(\mathbf{x},\mathbf{z})]\ \forall \mathbf{x} \in \mathcal{D}_\mathbf{x}, \mathbf{z} \in \mathcal{D}_\mathbf{z}, t \ge 1$ holds with probability $\ge 1-\delta$ where

$$\begin{aligned}l_{t-1}(\mathbf{x},\mathbf{z}) &\triangleq \mu_{t-1}(\mathbf{x},\mathbf{z}) - \beta_t^{1/2}\sigma_{t-1}(\mathbf{x},\mathbf{z})\\ u_{t-1}(\mathbf{x},\mathbf{z}) &\triangleq \mu_{t-1}(\mathbf{x},\mathbf{z}) + \beta_t^{1/2}\sigma_{t-1}(\mathbf{x},\mathbf{z})\ .\end{aligned} \quad (3)$$

As the above lemma holds for both finite and continuous $\mathcal{D}_x$ and $\mathcal{D}_z$, it is used to analyse the regret in both cases. On the other hand, the confidence bound can be adopted to the Bayesian setting by changing only β_t following the work of Srinivas et al. (2010) as noted by Bogunovic et al. (2018).

Then, we exploit this confidence bound on the function evaluations (Lemma 1) to formulate a confidence bound of $V_\alpha(f(\mathbf{x},\mathbf{Z}))$ as follows.

Lemma 2. Similar to the definition of $f(\mathbf{x},\mathbf{Z})$, let $l_{t-1}(\mathbf{x},\mathbf{Z})$ and $u_{t-1}(\mathbf{x},\mathbf{Z})$ denote the random function over $\mathbf{x}$ where the randomness comes from the random variable $\mathbf{Z}$; l_{t-1} and u_{t-1} are defined in (3). Then, $\forall \mathbf{x} \in \mathcal{D}_\mathbf{x}, t \ge 1$,

$$\begin{aligned}V_\alpha(f(\mathbf{x},\mathbf{Z})) &\in I_{t-1}[V_\alpha(f(\mathbf{x},\mathbf{Z}))]\\ &\triangleq [V_\alpha(l_{t-1}(\mathbf{x},\mathbf{Z})), V_\alpha(u_{t-1}(\mathbf{x},\mathbf{Z}))]\end{aligned}$$

holds with probability $\ge 1-\delta$ for δ in Lemma 1, where $V_\alpha(l_{t-1}(\mathbf{x},\mathbf{Z}))$ and $V_\alpha(u_{t-1}(\mathbf{x},\mathbf{Z}))$ are defined as (1).

The proof is in Appendix A. Given the confidence bound $I_{t-1}[V_\alpha(f(\mathbf{x},\mathbf{Z}))] \triangleq [V_\alpha(l_{t-1}(\mathbf{x},\mathbf{Z})), V_\alpha(u_{t-1}(\mathbf{x},\mathbf{Z}))]$ in Lemma 2, we follow the the well-known "optimism in the face of uncertainty" principle to select $\mathbf{x}_t = \text{argmax}_{\mathbf{x}\in\mathcal{D}_\mathbf{x}} V_\alpha(u_{t-1}(\mathbf{x},\mathbf{Z}))$. This query selection strategy for $\mathbf{x}_t$ leads to an upper bound of $r(\mathbf{x}_t)$:

$$r(\mathbf{x}_t) \le V_\alpha(u_{t-1}(\mathbf{x}_t,\mathbf{Z})) - V_\alpha(l_{t-1}(\mathbf{x}_t,\mathbf{Z}))\ \forall t \ge 1 \quad (4)$$

which holds with probability $\ge 1-\delta$ for δ in Lemma 1, and is proved in Appendix B.

As our goal is $\lim_{T\to\infty} R_T/T = 0$, given the selected query $\mathbf{x}_t$, a reasonable query selection strategy of $\mathbf{z}_t$ should gather informative observations at $(\mathbf{x}_t,\mathbf{z}_t)$ that improves the confidence bound $I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))]$ (i.e., $I_t[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))]$ is a proper subset of $I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))]$ if $I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))] \neq \emptyset$) which can be viewed as the uncertainty of $V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))$.

Assume that there exists $\mathbf{z}_l \in \mathcal{D}_\mathbf{z}$ such that $l_{t-1}(\mathbf{x}_t,\mathbf{z}_l) = V_\alpha(l_{t-1}(\mathbf{x}_t,\mathbf{Z}))$ and $\mathbf{z}_u \in \mathcal{D}_\mathbf{z}$ such that $u_{t-1}(\mathbf{x}_t,\mathbf{z}_u) = V_\alpha(u_{t-1}(\mathbf{x}_t,\mathbf{Z}))$. Lemma 2 implies that $V_\alpha(f(\mathbf{x}_t,\mathbf{Z})) \in I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))] = [l_{t-1}(\mathbf{x}_t,\mathbf{z}_l), u_{t-1}(\mathbf{x}_t,\mathbf{z}_u)]$ with high probability. Hence, we may naïvely want to query for observations at $(\mathbf{x}_t,\mathbf{z}_l)$ and $(\mathbf{x}_t,\mathbf{z}_u)$ to reduce $I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))]$. However, these observations may not always reduce $I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))]$. The insight is that $I_{t-1}[V_\alpha(f(\mathbf{x}_t,\mathbf{Z}))]$ changes (i.e., shrinks) when either of its boundary values (i.e., $l_{t-1}(\mathbf{x}_t,\mathbf{z}_l)$ or $u_{t-1}(\mathbf{x}_t,\mathbf{z}_u)$) changes.

Consider $u_{t-1}(\mathbf{x}_t, \mathbf{z}_u)$ and finite $\mathcal{D}_\mathbf{z}$ as an example, since $u_{t-1}(\mathbf{x}_t, \mathbf{z}_u) = V_\alpha(u_{t-1}(\mathbf{x}_t, \mathbf{Z}))$, a natural cause for the change in $u_{t-1}(\mathbf{x}_t, \mathbf{z}_u)$ is when $\mathbf{z}_u$ changes. This happens if there exists $\mathbf{z}' \neq \mathbf{z}_u$ such that the *ordering* of $u_{t-1}(\mathbf{x}_t, \mathbf{z}')$ relative to $u_{t-1}(\mathbf{x}_t, \mathbf{z}_u)$ changes given more observations. Thus, observations that are capable of reducing $I_{t-1}[V_\alpha(f(\mathbf{x}_t, \mathbf{Z}))]$ should be able to *change the relative ordering* in this case. We construct the following counterexample where observations at $\mathbf{z}_u$ (and $\mathbf{z}_l$) are not able to change the relative ordering, so they do not reduce $I_{t-1}[V_\alpha(f(\mathbf{x}_t, \mathbf{Z}))]$.

Example 1. This example is described by Fig. 1. We reduce notational clutter by removing $\mathbf{x}_t$ and t since they are fixed in this example, i.e., we use $f(\mathbf{z})$, $f(\mathbf{Z})$, and $l(\mathbf{z})$ to denote $f(\mathbf{x}_t, \mathbf{z})$, $f(\mathbf{x}_t, \mathbf{Z})$, and $l_{t-1}(\mathbf{x}_t, \mathbf{z})$ respectively. We condition on the event $f(\mathbf{z}) \in I[f(\mathbf{z})] \triangleq [l(\mathbf{z}), u(\mathbf{z})]$ for all $\mathbf{z} \in \mathcal{D}_\mathbf{z}$ which occurs with probability $\geq 1 - \delta$ in Lemma 1. In this example, $\mathbf{z}_l = \mathbf{z}_1$ and $l(\mathbf{z}_1) = u(\mathbf{z}_1)$, so there is no uncertainty in $f(\mathbf{z}_l) = f(\mathbf{z}_1)$. Similarly, there is no uncertainty in $f(\mathbf{z}_u) = f(\mathbf{z}_2)$. Thus, new observations at $\mathbf{z}_l$ and $\mathbf{z}_u$ change neither $l(\mathbf{z}_l)$ nor $u(\mathbf{z}_u)$, so these observations do not reduce the confidence bound $I[V_{\alpha=0.4}(f(\mathbf{Z}))] = [l(\mathbf{z}_l), u(\mathbf{z}_u)]$ (plotted as the double-headed arrow in Fig. 1b). In fact, to reduce $I[V_{\alpha=0.4}(f(\mathbf{Z}))]$, we should gather new observations at $\mathbf{z}_0$ which potentially change the ordering of $u(\mathbf{z}_0)$ relative to $u(\mathbf{z}_2)$ (which is $u(\mathbf{z}_u)$ without new observations). For example, after getting new observations at $\mathbf{z}_0$, if $u(\mathbf{z}_0)$ is improved to be in the white region between A and B ($u(\mathbf{z}_0) > u(\mathbf{z}_2)$ in Fig. 1b changes to $u(\mathbf{z}_0) < u(\mathbf{z}_2)$ in Fig. 1c), then $I[V_{\alpha=0.4}(f(\mathbf{Z}))]$ is reduced to $[l(\mathbf{z}_1), u(\mathbf{z}_0)]$ because now $\mathbf{z}_u = \mathbf{z}_0$. Thus, as the confidence bound $I[f(\mathbf{z}_0)]$ is shortened with more and more observations at $\mathbf{z}_0$, the confidence bound $I[V_{\alpha=0.4}(f(\mathbf{Z}))]$ reduces (the white region representing the uncertainty of $V_{\alpha=0.4}(f(\mathbf{Z}))$ in Fig. 1 is 'laced up').

In the next section, we define a property of $\mathbf{z}_0$ in Example 1 and prove the existence of $\mathbf{z}$'s with this property. Then, we prove that along with the optimistic selection of $\mathbf{x}_t$, the selection of $\mathbf{z}_t$ such that it satisfies this property leads to a no-regret algorithm.

3.2. Lacing Value (LV) and the Query Selection Strategy for $\mathbf{z}_t$

We note that in Example 1, as long as the confidence bound of the function evaluation at $\mathbf{z}_0$ contains the confidence bound of VaR, observations at $\mathbf{z}_0$ can reduce the confidence bound of VaR. We name the values of $\mathbf{z}$ satisfying this property as *lacing values* (LV) since observations at LV 'laces up' the confidence bound of VaR:

Definition 1 (Lacing values). *Lacing values* (LV) with respect to $\mathbf{x} \in \mathcal{D}_\mathbf{x}$ and $t \geq 1$ are $\mathbf{z}_{\text{LV}} \in \mathcal{D}_\mathbf{z}$ that satisfies $l_{t-1}(\mathbf{x}, \mathbf{z}_{\text{LV}}) \leq V_\alpha(l_{t-1}(\mathbf{x}, \mathbf{Z})) \leq V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z})) \leq u_{t-1}(\mathbf{x}, \mathbf{z}_{\text{LV}})$, equivalently, $I_{t-1}[V_\alpha(f(\mathbf{x}, \mathbf{Z}))] \subset [l_{t-1}(\mathbf{x}, \mathbf{z}_{\text{LV}}), u_{t-1}(\mathbf{x}, \mathbf{z}_{\text{LV}})]$.

Recall that the support $\mathcal{D}_\mathbf{z}$ of $\mathbf{Z}$ is defined as the smallest closed subset $\mathcal{D}_\mathbf{z}$ of $\mathbb{R}^{d_z}$ such that $P(\mathbf{Z} \in \mathcal{D}_\mathbf{z}) = 1$ (e.g., $\mathcal{D}_\mathbf{z}$ is a finite subset of $\mathbb{R}^{d_z}$ and $\mathcal{D}_\mathbf{z} = \mathbb{R}^{d_z}$). The following theorem guarantees the existence of lacing values and is proved in Appendix C.

Theorem 1 (Existence of lacing values)**.** $\forall \alpha \in (0, 1), \forall \mathbf{x} \in \mathcal{D}_\mathbf{x}, \forall t \geq 1$, there exists a lacing value in $\mathcal{D}_\mathbf{z}$ with respect to $\mathbf{x}$ and t.

Corollary 1.1. Lacing values with respect to $\mathbf{x} \in \mathcal{D}_\mathbf{x}$ and $t \geq 1$ are in $\mathcal{Z}_l^{\leq}$ and $\mathcal{Z}_u^{\geq}$ (hence, their intersection) where $\mathcal{Z}_l^{\leq} \triangleq \{\mathbf{z} \in \mathcal{D}_\mathbf{z} : l_{t-1}(\mathbf{x}, \mathbf{z}) \leq V_\alpha(l_{t-1}(\mathbf{x}, \mathbf{Z}))\}$ and $\mathcal{Z}_u^{\geq} \triangleq \{\mathbf{z} \in \mathcal{D}_\mathbf{z} : u_{t-1}(\mathbf{x}, \mathbf{z}) \geq V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))\}$.

As a special case, when $\mathbf{z}_l = \mathbf{z}_u$, $I_{t-1}[V_\alpha(f(\mathbf{x}, \mathbf{Z}))] = I_{t-1}[f(\mathbf{x}, \mathbf{z}_l)]$ which means $\mathbf{z}_l = \mathbf{z}_u$ is an LV. Based on Theorem 1, we can always select $\mathbf{z}_t$ as an LV w.r.t $\mathbf{x}_t$ defined in Definition 1. This strategy for the selection of $\mathbf{z}_t$, together with the selection of $\mathbf{x}_t = \text{argmax}_{\mathbf{x} \in \mathcal{D}_\mathbf{x}} V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ (Section 3.1), completes our algorithm: VaR *Upper Confidence Bound* (V-UCB) (Algorithm 1).

Upper Bound on Regret. As a result of the selection strategies for $\mathbf{x}_t$ and $\mathbf{z}_t$, our V-UCB algorithm enjoys the following upper confidence bound on its instantaneous regret (proved in Appendix D):

Lemma 3. By selecting $\mathbf{x}_t$ as a maximizer of $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ and selecting $\mathbf{z}_t$ as an LV w.r.t $\mathbf{x}_t$, the instantaneous regret is upper-bounded by:

$$r(\mathbf{x}_t) \leq 2\beta_t^{1/2} \sigma_{t-1}(\mathbf{x}_t, \mathbf{z}_t) \; \forall t \geq 1$$

with probability $\geq 1 - \delta$ for δ in Lemma 1.

We assume that the $k_{(\mathbf{x},\mathbf{z}),(\mathbf{x},\mathbf{z})} \leq 1$ for all $(\mathbf{x}, \mathbf{z}) \in \mathcal{D}_\mathbf{x} \times \mathcal{D}_\mathbf{z}$. Then, Lemma 3 and Lemma 5.4 from Srinivas et al. (2010) imply that the cumulative regret of our algorithm is bounded (Appendix E): $R_T \leq \sqrt{C_1 T \beta_T \gamma_T}$ where $C_1 \triangleq 8/\log(1 + \sigma_n^{-2})$, and γ_T is the maximum information gain about f that can be obtained from any set of T observations. The work of Srinivas et al. (2010) has analyzed γ_T for several commonly used kernels such as SE and Matérn kernels, and has shown that for these kernels, the upper confidence bound on R_T grows sub-linearly. This implies that our algorithm is *no-regret* because $\lim_{T \to \infty} R_T / T = 0$.

Following the work of Bogunovic et al. (2018), at the T-th iteration of V-UCB, we can recommend $\mathbf{x}_{t_*(T)}$ as an estimate of the maximizer $\mathbf{x}_*$ of $V_\alpha(f(\mathbf{x}, \mathbf{Z}))$, where $t_*(T) \triangleq \text{argmax}_{t \in \{1,\ldots,T\}} V_\alpha(l_{t-1}(\mathbf{x}_t, \mathbf{Z}))$. Then, the instantaneous regret $r(\mathbf{x}_{t_*(T)})$ is shown to be upper-bounded by $\sqrt{C_1 \beta_T \gamma_T / T}$ with high probability (see Appendix F). In our experiments in Section 4, we recommend

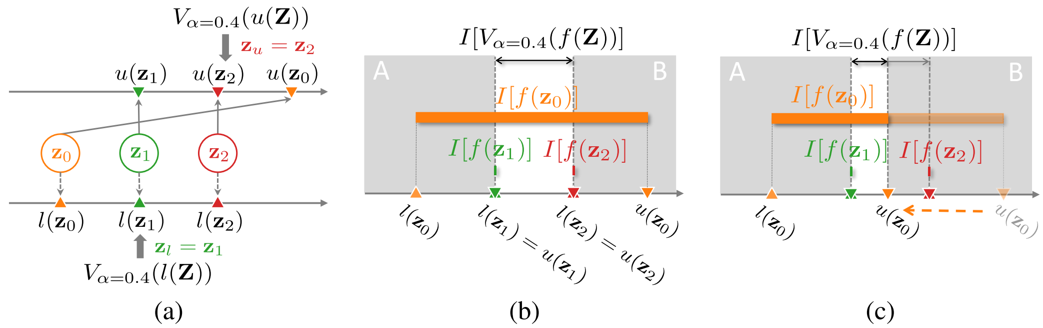


Figure 1. A counterexample against selecting $\mathbf{z}_u$ and $\mathbf{z}_l$ as input queries. Here $\mathbf{z}$ follows a discrete uniform distribution over $\mathcal{D}_{\mathbf{z}} \triangleq \{\mathbf{z}_0, \mathbf{z}_1, \mathbf{z}_2\}$. (a) shows the mappings of $\mathbf{z}$ to the upper bound $u(\mathbf{z})$ and lower bound $l(\mathbf{z})$. The VAR at $\alpha = 0.4$ of $u(\mathbf{Z})$ and $l(\mathbf{Z})$ are $u(\mathbf{z}_2)$ and $l(\mathbf{z}_1)$, respectively, i.e., $\mathbf{z}_u = \mathbf{z}_2$ and $\mathbf{z}_l = \mathbf{z}_1$. (b) shows the values of $l(\mathbf{z})$ and $u(\mathbf{z})$ for all $\mathbf{z}$ on the same axis, as well as the confidence bounds of $f(\mathbf{z})$ and $V_\alpha(f(\mathbf{Z}))$. The gray areas A and B indicate the intervals of values $\omega \in \mathbb{R}$ where $\omega \le l(\mathbf{z}_l) = l(\mathbf{z}_1)$ and $\omega \ge u(\mathbf{z}_u) = u(\mathbf{z}_2)$, respectively. (c) shows a hypothetical scenario when $I[f(\mathbf{z}_0)]$ is shortened/'laced-up' with observations at $\mathbf{z}_0$.

Algorithm 1 The V-UCB Algorithm

1: **Input:** $\mathcal{D}_{\mathbf{x}}$, $\mathcal{D}_{\mathbf{z}}$, prior $\mu_0 = 0, \sigma_0, k$
2: **for** $i = 1, 2, \ldots$ **do**
3: Select $\mathbf{x}_t = \text{argmax}_{\mathbf{x} \in \mathcal{D}_{\mathbf{x}}} V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$
4: Select $\mathbf{z}_t$ as a *lacing value* w.r.t. $\mathbf{x}_t$ (Definition 1)
5: Obtain observation $y_t \triangleq f(\mathbf{x}_t, \mathbf{z}_t) + \epsilon_t$
6: Update the GP posterior belief to obtain μ_t and σ_t
7: **end for**

$\text{argmax}_{\mathbf{x} \in \mathcal{D}_T} V_\alpha(\mu_{t-1}(\mathbf{x}, \mathbf{Z}))$ (where $\mu_{t-1}(\mathbf{x}, \mathbf{Z})$ is a random function defined in the same manner as $f(\mathbf{x}, \mathbf{Z})$) as an estimate of $\mathbf{x}_*$ due to its empirical convergence.

Computational Complexity. To compare our computational complexity with that of the ρKG and ρKG^{apx} algorithms in Cakmak et al. (2020), we exclude the training of the GP model (line 6 of Algorithm 1) and assume that $\mathcal{D}_{\mathbf{z}}$ is finite. Then, the time complexity of V-UCB is dominated by that of the selection of $\mathbf{x}_t$ (line 3 of Algorithm 1) which includes the time complexity $\mathcal{O}(|\mathcal{D}_{\mathbf{z}}||\mathcal{D}_{t-1}|^2)$ for the GP prediction at $\{\mathbf{x}\} \times \mathcal{D}_{\mathbf{z}}$, and $\mathcal{O}(|\mathcal{D}_{\mathbf{z}}| \log |\mathcal{D}_{\mathbf{z}}|)$ for the sorting of $u_{t-1}(\mathbf{x}, \mathcal{D}_{\mathbf{z}})$ and searching of VAR. Hence, our overall complexity is $\mathcal{O}(n|\mathcal{D}_{\mathbf{z}}|\,(|\mathcal{D}_{t-1}|^2 + \log|\mathcal{D}_{\mathbf{z}}|))$, where n is the number of iterations to maximize $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ (line 3 of Algorithm 1). This time complexity of V-UCB is more computationally efficient than ρKG and its variant with approximation ρKG^{apx} whose time complexities are shown in the work of Cakmak et al. (2020) as $\mathcal{O}(n_{\text{out}} n_{\text{in}} K |\mathcal{D}_{\mathbf{z}}|\,(|\mathcal{D}_{t-1}|^2 + |\mathcal{D}_{\mathbf{z}}||\mathcal{D}_{t-1}| + |\mathcal{D}_{\mathbf{z}}|^2 + M|\mathcal{D}_{\mathbf{z}}|))$ and $\mathcal{O}(n_{\text{out}} |\mathcal{D}_{t-1}| K |\mathcal{D}_{\mathbf{z}}|\,(|\mathcal{D}_{t-1}|^2 + |\mathcal{D}_{\mathbf{z}}||\mathcal{D}_{t-1}| + |\mathcal{D}_{\mathbf{z}}|^2 + M|\mathcal{D}_{\mathbf{z}}|))$, respectively ($n_{\text{out}}$ and n_{in} are the numbers of iterations for the outer and inner optimization procedures, respectively; K is the number of fantasy GP models required for ρKG's and ρKG^{apx}'s one-step lookahead, and M is the number of functions sampled from the GP posterior belief).

3.3. On the Selection of $\mathbf{z}_t$

Although Algorithm 1 is guaranteed to be no-regret with any choice of LV as $\mathbf{z}_t$, we would like to select the LV that can reduce a large amount of the uncertainty of $V_\alpha(f(\mathbf{x}_t, \mathbf{Z}))$. However, relying on the information gain measure or the knowledge gradient method often incurs the expensive one-step lookahead. Therefore, we use a simple heuristic by choosing the LV $\mathbf{z}_{\text{LV}}$ with the maximum probability mass (or probability density if $\mathbf{Z}$ is continuous) of $\mathbf{z}_{\text{LV}}$. We motivate this heuristic using an example with $\alpha = 0.2$, i.e., $V_{\alpha=0.2}(f(\mathbf{x}_t, \mathbf{Z})) = \inf\{\omega : P(f(\mathbf{x}_t, \mathbf{Z}) \le \omega) \ge 0.2\}$. Suppose $\mathcal{D}_{\mathbf{z}}$ is finite and there are 2 LV's $\mathbf{z}_0$ and $\mathbf{z}_1$ with $P(\mathbf{z}_0) \ge 0.2$ and $P(\mathbf{z}_1) = 0.01$. Then, because $P(f(\mathbf{x}_t, \mathbf{Z}) \le f(\mathbf{x}_t, \mathbf{z}_0)) \ge P(\mathbf{z}_0) \ge 0.2$, it follows that $V_{\alpha=0.2}(f(\mathbf{x}_t, \mathbf{Z})) \le f(\mathbf{x}_t, \mathbf{z}_0)$, i.e., querying $\mathbf{z}_0$ at $\mathbf{x}_t$ gives us information about an explicit upper bound on $V_{\alpha=0.2}(f(\mathbf{x}_t, \mathbf{Z}))$ to reduce its uncertainty. In contrast, this cannot be achieved by querying $\mathbf{z}_1$ due to its low probability mass. This simple heuristic can also be implemented when $\mathbf{Z}$ is a continuous random variable which we will introduce in Section 3.5.

Remark 1. Although we assume that we can select both $\mathbf{x}_t$ and $\mathbf{z}_t$ during our algorithm, Corollary 1.1 also gives us some insights about the scenario where we cannot select $\mathbf{z}_t$. In this case, in each iteration t, we select $\mathbf{x}_t$ while $\mathbf{z}_t$ is randomly sampled by the environment following the distribution of $\mathbf{Z}$. Next, we observe both $\mathbf{z}_t$ and $y_{(\mathbf{x}_t, \mathbf{z}_t)}$ and then update the GP posterior belief of f. Of note, Corollary 1.1 has shown that all LV lie in the set $\mathcal{Z}_l^{\le}$. However, the probability of this set is usually small, because $P(\mathbf{Z} \in \mathcal{Z}_l^{\le}) \le \alpha$ and small values of α are often used by real-world applications to manage risks. Thus, there is a small probability that the sampled $\mathbf{z}_t$ is an LV. As a result, we suggest sampling a large number of $\mathbf{z}_t$'s from the environment to increase the chance that an LV is sampled. On the other hand, the small probability of sampling an LV motivates the need for us to select $\mathbf{z}_t$.

3.4. V-UCB Approaches STABLEOPT as $\alpha \to 0^+$

Recall that the objective of adversarially robust BO is to find $\mathbf{x} \in \mathcal{D}_{\mathbf{x}}$ that maximizes $\min_{\mathbf{z} \in \mathcal{D}_{\mathbf{z}}} f(\mathbf{x}, \mathbf{z})$ (Bogunovic et al., 2018) by iteratively specifying input query $(\mathbf{x}_t, \mathbf{z}_t)$ to collect noisy observations $y_{\mathbf{x}_t, \mathbf{z}_t}$. It is different from BO of VAR since its $\mathbf{z}$ is not random but selected by an adversary who aims to minimize the function evaluation. The work of Bogunovic et al. (2018) has proposed a no-regret algorithm for this setting named STABLEOPT, which selects

$$\begin{aligned} \mathbf{x}_t &= \arg\max_{\mathbf{x} \in \mathcal{D}_x} \min_{\mathbf{z} \in \mathcal{D}_z} u_{t-1}(\mathbf{x}, \mathbf{z}) \ , \\ \mathbf{z}_t &= \arg\min_{\mathbf{z} \in \mathcal{D}_z} l_{t-1}(\mathbf{x}_t, \mathbf{z}) \end{aligned} \tag{5}$$

where u_{t-1} and l_{t-1} are defined in (3).

At first glance, BO of VAR and adversarially robust BO are seemingly different problems because $\mathbf{Z}$ is a random variable in the former but not in the latter. However, based on our key observation on the connection between the minimum value of a continuous function $h(\mathbf{w})$ and the VAR of the random variable $h(\mathbf{W})$ in the following theorem, these two problems and their solutions are connected as shown in Corollary 2.1, and 2.2.

Theorem 2. Let $\mathbf{W}$ be a random variable with the support $\mathcal{D}_w \subset \mathbb{R}^{d_w}$ and dimension d_w. Let h be a continuous function mapping from $\mathbf{w} \in \mathcal{D}_w$ to $\mathbb{R}$. Then, $h(\mathbf{W})$ denotes the random variable whose realization is the function h evaluation at a realization $\mathbf{w}$ of $\mathbf{W}$. Suppose $h(\mathbf{w})$ has a minimizer $\mathbf{w}_{\min} \in \mathcal{D}_w$, then $\lim_{\alpha \to 0^+} V_\alpha(h(\mathbf{W})) = h(\mathbf{w}_{\min})$.

Corollary 2.1. Adversarially robust BO which finds $\text{argmax}_{\mathbf{x}} \min_{\mathbf{z}} f(\mathbf{x}, \mathbf{z})$ can be cast as BO of VAR by letting (a) α approach 0 from the right and (b) $\mathcal{D}_{\mathbf{z}}$ be the support of the environmental random variable $\mathbf{Z}$, i.e., $\text{argmax}_{\mathbf{x}} \lim_{\alpha \to 0^+} V_\alpha(f(\mathbf{x}, \mathbf{Z}))$.

Interestingly, from Theorem 2, we observe that $\mathcal{Z}_l^{\leq}$ in Corollary 1.1 approaches the set of minimizers $\text{argmin}_{\mathbf{z} \in \mathcal{D}_{\mathbf{z}}} l_{t-1}(\mathbf{x}_t, \mathbf{z})$ as $\alpha \to 0^+$. Corollary 2.2 below shows that LV w.r.t $\mathbf{x}_t$ becomes a minimizer of $l_{t-1}(\mathbf{x}_t, \mathbf{z})$ which is same as the selected $\mathbf{z}_t$ of STABLEOPT in (5).

Corollary 2.2. The STABLEOPT solution to adversarially robust BO selects the same input query as that selected by the V-UCB solution to the corresponding BO of VAR in Corollary 2.1.

The proof of Theorem 2 and its two corollaries are shown in Appendix G. We note that V-UCB is also applicable to the optimization of $V_\alpha(f(\mathbf{x}, \mathbf{Z}))$ where the distribution of $\mathbf{Z}$ is a conditional distribution given $\mathbf{x}$. For example, in robotics, if there exists noise/perturbation in the control, an optimization problem of interest is $V_\alpha(f(\mathbf{x} + \boldsymbol{\xi}(\mathbf{x})))$ where $\boldsymbol{\xi}(\mathbf{x})$ is a random perturbation that depends on $\mathbf{x}$.

3.5. Implementation of V-UCB with Continuous Random Variable Z

The V-UCB algorithm involves two steps: selecting $\mathbf{x}_t = \text{argmax}_{\mathbf{x} \in \mathcal{D}_{\mathbf{x}}} V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ and selecting $\mathbf{z}_t$ as the LV $\mathbf{z}_{\text{LV}}$ with the largest probability mass (or probability density). When $\mathcal{D}_{\mathbf{z}}$ is finite, given $\mathbf{x}$, $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ can be computed exactly. The gradient of $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ with respect to $\mathbf{x}$ can be obtained easily (e.g., using automatic differentiation provided in the Tensorflow library (Abadi et al., 2015)) to aid the selection of $\mathbf{x}_t$. In this case, the latter step can also be performed by constructing the set of all LV (checking the condition in the Definition 1 for all $\mathbf{z} \in \mathcal{D}_{\mathbf{z}}$) and choosing the LV $\mathbf{z}_{\text{LV}}$ with the largest probability mass.

Estimation of VAR. When $\mathbf{Z}$ is a continuous random variable, estimating VAR by an ordered set of samples (e.g., in Cakmak et al. (2020)) may require a prohibitively large number of samples, especially for small values of α. Thus, we employ the following popular pinball (or tilted absolute value) function in quantile regression (Koenker & Bassett, 1978) to estimate VAR:

$$\rho_\alpha(w) \triangleq \begin{cases} \alpha w & \text{if } w \geq 0 \ , \\ (\alpha - 1) w & \text{if } w < 0 \end{cases}$$

where $w \in \mathbb{R}$. In particular, to estimate $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ as $\nu \in \mathbb{R}$, we find ν that minimizes:

$$\mathbb{E}_{\mathbf{z} \sim p(\mathbf{Z})} [\rho_\alpha(u_{t-1}(\mathbf{x}, \mathbf{z}) - \nu)] \ . \tag{6}$$

A well-known example is $\alpha = 0.5$, then ρ_α is the absolute value function and the optimal ν is the median. The loss in (6) can be optimized using stochastic gradient descent with a random batch of samples of $\mathbf{Z}$ at each optimization iteration.

Maximization of $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$. Unfortunately, to maximize $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ over $\mathbf{x} \in \mathcal{D}_{\mathbf{x}}$, there is no gradient of $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ with respect to $\mathbf{x}$ under the above approach. This situation resembles the BO problem where there is no gradient information, but only noisy observations at input queries. Unlike BO, the observation (samples of $u_{t-1}(\mathbf{x}, \mathbf{Z})$ at $\mathbf{x}$) is not costly. Therefore, we propose the *local neural surrogate optimization* (LNSO) algorithm to find $\text{argmax}_{\mathbf{x} \in \mathcal{D}_{\mathbf{x}}} V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ which is visualized in Fig. 2. Suppose the optimization is initialized at $\mathbf{x} = \mathbf{x}^{(0)}$, instead of maximizing $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ (whose gradient w.r.t. $\mathbf{x}$ is unknown), we maximize a surrogate function $g(\mathbf{x}, \boldsymbol{\theta}^{(0)})$ (modeled by a neural network) that approximates $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ well in a local region of $\mathbf{x}^{(0)}$, e.g., a ball $\mathcal{B}(\mathbf{x}^{(0)}, r)$ of radius r in Fig. 2. We obtain such parameters $\boldsymbol{\theta}^{(0)}$ by minimizing the following loss function:

$$\begin{aligned} &\mathcal{L}_g(\boldsymbol{\theta}, \mathbf{x}^{(0)}) \\ &\triangleq \mathbb{E}_{\mathbf{x} \in \mathcal{B}(\mathbf{x}^{(0)}, r)} \mathbb{E}_{\mathbf{Z} \sim p(\mathbf{Z})} \left[\rho_\alpha(u_{t-1}(\mathbf{x}, \mathbf{z}) - g(\mathbf{x}; \boldsymbol{\theta})) \right] \end{aligned} \tag{7}$$

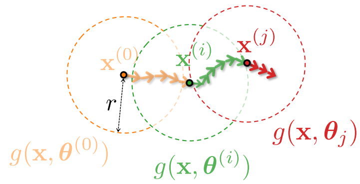


Figure 2. Plot of a hypothetical optimization path (as arrows) of LNSO initialized at $\mathbf{x}^{(0)}$. Input $\mathbf{x}$ is 2-dimensional. The boundary of a ball $\mathcal{B}$ of radius r is plotted as a dotted circle. When the updated $\mathbf{x}$ moves out of $\mathcal{B}$, the center of $\mathcal{B}$ and $\boldsymbol{\theta}$ are updated.

where the expectation $\mathbb{E}_{\mathbf{x}\in\mathcal{B}(\mathbf{x}^{(0)},r)}$ is taken over a uniformly distributed $\mathbf{X}$ in $\mathcal{B}(\mathbf{x}^{(0)}, r)$. Minimizing (7) can be performed with stochastic gradient descent. If maximizing $g(\mathbf{x}, \boldsymbol{\theta}^{(0)})$ leads to a value $\mathbf{x}^{(i)} \notin \mathcal{B}(\mathbf{x}^{(0)}, r)$ (Fig. 2), we update the local region to be centered at $\mathbf{x}^{(i)}$ ($\mathcal{B}(\mathbf{x}^{(i)}, r)$) and find $\boldsymbol{\theta}^{(i)} = \text{argmin}_{\boldsymbol{\theta}} \mathcal{L}_g(\boldsymbol{\theta}, \mathbf{x}^{(i)})$ such that $g(\mathbf{x}, \boldsymbol{\theta}^{(i)})$ approximates $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ well $\forall \mathbf{x} \in \mathcal{B}(\mathbf{x}^{(i)}, r)$. Then, $\mathbf{x}^{(i)}$ is updated by maximizing $g(\mathbf{x}, \boldsymbol{\theta}^{(i)})$ for $\mathbf{x} \in \mathcal{B}(\mathbf{x}^{(i)}, r)$. The complete algorithm is described in Appendix H.

We prefer a small value of r so that the ball $\mathcal{B}$ is small. In such case, $V_\alpha(u_{t-1}(\mathbf{x}, \mathbf{Z}))$ for $\mathbf{x} \in \mathcal{B}$ can be estimated well with a small neural network $g(\mathbf{x}, \boldsymbol{\theta})$ whose training requires a small number of iterations.

Search of Lacing Values. Given a continuous random variable $\mathbf{Z}$, to find an LV w.r.t $\mathbf{x}_t$ in line 4 of Algorithm 1, i.e., to find a $\mathbf{z}$ satisfying $d_u(\mathbf{z}) \triangleq u_{t-1}(\mathbf{x}_t, \mathbf{z}) - V_\alpha(u_{t-1}(\mathbf{x}_t, \mathbf{Z})) \geq 0$ and $d_l(\mathbf{z}) \triangleq V_\alpha(l_t(\mathbf{x}_t, \mathbf{Z})) - l_{t-1}(\mathbf{x}_t, \mathbf{z}) \geq 0$, we choose a $\mathbf{z}$ that minimizes

$$\mathcal{L}_{\text{LV}}(\mathbf{z}) \triangleq \text{ReLU}(-d_u(\mathbf{z})) + \text{ReLU}(-d_l(\mathbf{z})) \quad (8)$$

where $\text{ReLU}(\omega) = \max(\omega, 0)$ is the rectified linear unit function ($\omega \in \mathbb{R}$). To include the heuristic in Section 3.3 which selects the LV with the highest probability density, we find $\mathbf{z}$ that minimizes

$$\mathcal{L}_{\text{LV-P}}(\mathbf{z}) \triangleq \mathcal{L}_{\text{LV}}(\mathbf{z}) - \mathbb{I}_{d_u(\mathbf{z})\geq 0}\mathbb{I}_{d_l(\mathbf{z})\geq 0}\, p(\mathbf{z})$$

where $\mathcal{L}_{\text{LV}}(\mathbf{z})$ is defined in (8); $p(\mathbf{z})$ is the probability density; $\mathbb{I}_{d_u(\mathbf{z})\geq 0}$ and $\mathbb{I}_{d_l(\mathbf{z})\geq 0}$ are indicator functions.

4. Experiments

In this section, we empirically evaluate the performance of V-UCB. The work of Cakmak et al. (2020) has motivated the use of the approximated variant of their algorithm ρKG^{apx} over its original version ρKG by showing that ρKG^{apx} achieves comparable empirical performances to ρKG while incurring much less computational cost. Furthermore, ρKG^{apx} has been shown to significantly outperform other competing algorithms (Cakmak et al., 2020). Therefore, we choose ρKG^{apx} as the main baseline to empirically compare with V-UCB. The experiments using ρKG^{apx} is performed by adding new objective functions to the existing implementation of Cakmak et al. (2020) at `https://github.com/saitcakmak/BoRisk`.

Regarding V-UCB, when $\mathcal{D}_{\mathbf{z}}$ is finite and the distribution of $\mathbf{Z}$ is not uniform, we perform V-UCB by selecting $\mathbf{z}_t$ as an LV at random, labeled as *V-UCB Unif*, and by selecting $\mathbf{z}_t$ as the LV with the maximum probability mass, labeled as *V-UCB Prob*.

The performance metric is defined as $V_\alpha(f(\mathbf{x}_*, \mathbf{Z})) - V_\alpha(f(\tilde{\mathbf{x}}, \mathbf{Z}))$ where $\tilde{\mathbf{x}}$ is the recommended input. The evaluation of VaR is described in Section 3.5. The recommended input is $\text{argmax}_{\mathbf{x}\in\mathcal{D}_T} V_\alpha(\mu_{t-1}(\mathbf{x}, \mathbf{Z}))$ for V-UCB, and $\text{argmin}_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}} \mathbb{E}_{t-1}[V_\alpha(f(\mathbf{x}, \mathbf{Z}))]$ for ρKG^{apx} (Cakmak et al., 2020), where $\mathbb{E}_{t-1}$ is the conditional expectation over the unknown f given the observations $\mathbf{y}_{\mathcal{D}_{t-1}}$ (approximated by a finite set of functions sampled from the GP posterior belief).[2] We repeat each experiment 10 times with different random initial observations $\mathbf{y}_{\mathcal{D}_0}$ and plot both the mean (as lines) and the 70% confidence interval (as shaded areas) of the $\log 10$ of the performance metric. The detailed descriptions of experiments and the comparison with the strategy of randomly selecting input queries are deferred to Appendix I.

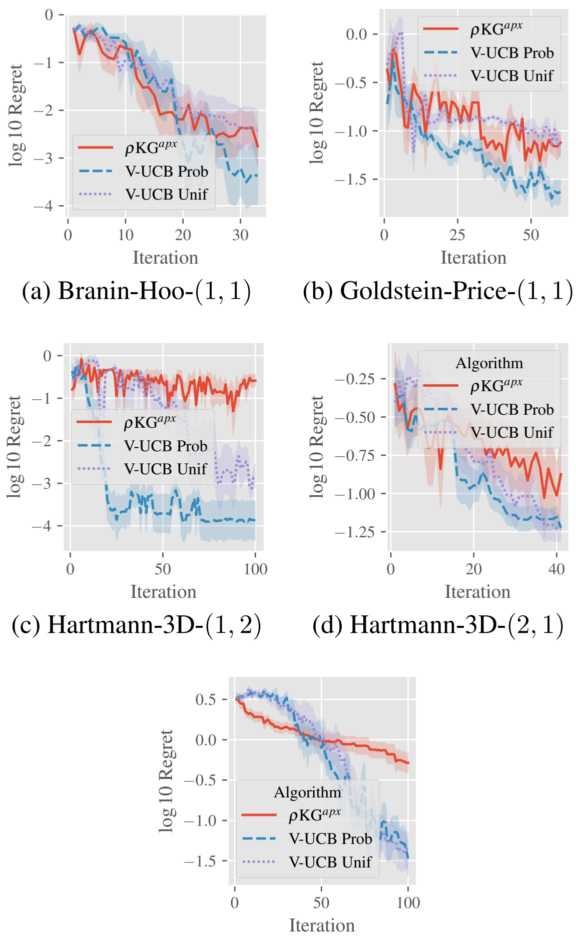
4.1. Synthetic Benchmark Functions

The experiments with discrete $\mathcal{D}_{\mathbf{z}}$ are conducted with Branin-Hoo-$(1,1)$, Goldstein-Price-$(1,1)$, Hartmann-3D-$(1,2)$, Hartmann-3D-$(2,1)$, and Hartmann-6D-$(5,1)$ where the tuple (d_x, d_z) represents the dimensions of $\mathbf{x}$ and $\mathbf{z}$. The noise variance σ_n^2 is set to 0.01. The risk level α is 0.1. We observe in Fig. 3 that V-UCB Unif is on par with ρKG^{apx} in optimizing Branin-Hoo-$(1,1)$ and Goldstein-Price-$(1,1)$, and outperforms ρKG^{apx} in optimizing Hartmann-3D-$(1,2)$, Hartmann-3D-$(2,1)$, and Hartmann-6D-$(5,1)$. On the other hand, V-UCB Prob is able to exploit the probability distribution of $\mathbf{Z}$ to outperform V-UCB Unif in optimizing Branin-Hoo-$(1,1)$, Goldstein-Price-$(1,1)$, Hartmann-3D-$(1,2)$, and Hartmann-3D-$(2,1)$. The unsatisfactory performance of ρKG^{apx} in some experiments may be attributed to its approximation of the inner optimization problem in the acquisition function (Cakmak et al., 2020), and the approximation of VaR using samples of $\mathbf{Z}$ and function samples of the GP posterior belief.

The experiments with continuous $\mathcal{D}_{\mathbf{z}}$ are conducted with Branin-Hoo-$(1,1)$, Goldstein-Price-$(1,1)$, Hartmann-3D-$(1,2)$, and Hartmann-3D-$(2,1)$. In the implementation of LNSO, the local region is defined with $r = 0.1$. The surrogate function is a neural network of 2 hidden layers with

[2] While the work of Cakmak et al. (2020) considers a minimization problem of VaR, our work considers a maximization problem of VaR. Therefore, the objective functions for ρKG^{apx} are the negation of those for V-UCB. For V-UCB at risk level α, the risk level for ρKG^{apx} is $1 - \alpha$.

30 hidden neurons each and sigmoid activation functions. The results are shown in Fig. 4. We observe that V-UCB Prob outperforms ρKG^{apx} because of the approximation involved in ρKG^{apx} as explained in the above experiments with discrete $\mathcal{D}_{\mathbf{z}}$.



(a) Branin-Hoo-$(1,1)$ (b) Goldstein-Price-$(1,1)$

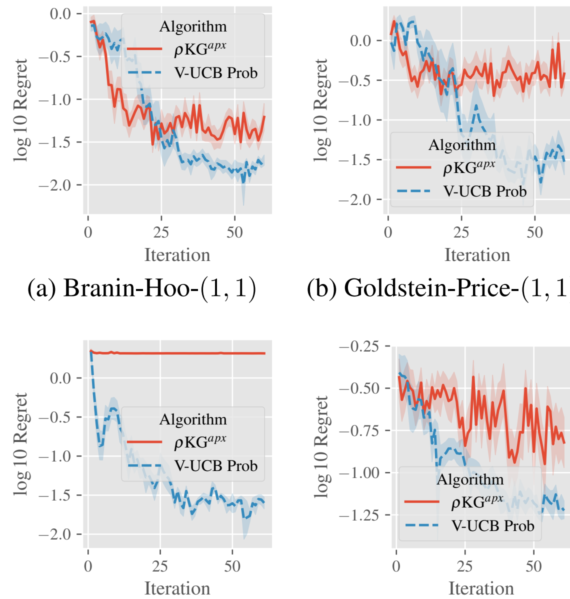
(c) Hartmann-3D-$(1,2)$ (d) Hartmann-3D-$(2,1)$

(e) Hartmann-6D-$(5,1)$

Figure 3. Synthetic benchmark functions with finite $\mathcal{D}_{\mathbf{z}}$.

4.2. Simulated Optimization Problems

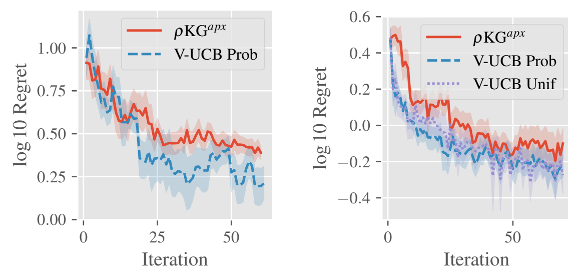
The first problem is portfolio optimization adopted by (Cakmak et al., 2020). There are $d_x = 3$ optimization variables (risk and trade aversion parameters, and the holding cost multiplier) and $d_z = 2$ environmental random variables (bid-ask spread and borrow cost). The variable $\mathbf{Z}$ follows a discrete uniform distribution with $|\mathcal{D}_{\mathbf{z}}| = 100$. Hence, there is no difference between V-UCB Unif and V-UCB Prob. Thus, we only report the results of the latter. The objective function is the posterior mean of a trained GP on the dataset in Cakmak et al. (2020) of size 3000 generated from CVXPortfolio. The noise variance σ_n^2 is set to 0.01. The risk level α is set to 0.2.



(a) Branin-Hoo-$(1,1)$ (b) Goldstein-Price-$(1,1)$

(c) Hartmann-3D-$(1,2)$ (d) Hartmann-3D-$(2,1)$

Figure 4. Synthetic benchmark functions with continuous $\mathcal{D}_{\mathbf{z}}$.



(a) Portfolio optimization problem $(3,2)$ (b) Simulated robot pushing task $(3,2)$

Figure 5. Simulated experiments.

The second problem is a simulated robot pushing task for which we use the implementation from the work of Wang & Jegelka (2017). The simulation is viewed as a 3-dimensional function $\mathbf{h}(r_x, r_y, t_p)$ returning the 2-D location of the pushed object, where $r_x, r_y \in [-5, 5]$ are the robot location and $t_p \in [1, 30]$ is the pushing duration. The objective is to minimize the distance to a fixed goal location $\mathbf{g} = (g_x, g_y)^\top$, i.e., the objective function of the maximization problem is $f_0(r_x, r_y, t_p) = -\|\mathbf{h}(r_x, r_y, t_p) - \mathbf{g}\|$. We assume that there are perturbations in specifying the robot location W_x, W_y whose support $\mathcal{D}_{\mathbf{z}}$ includes 64 equi-distant points in $[-1, 1]^2$ and whose probability mass is proportional to $\exp(-(W_x^2 + W_y^2)/0.4^2)$. It leads to a random objective function $f(r_x, r_y, t_p, W_x, W_y) \triangleq f_0(r_x + W_x, r_y + W_y, t_p)$. We aim to maximize the VaR of f which is more difficult than maximizing that of f_0. Moreover, a random noise following $\mathcal{N}(0, 0.01)$ is added to the evaluation of f.

The risk level α is set to 0.1.

The results are shown in Fig. 5. We observe that V-UCB outperforms ρKG^{apx} in both problems. Furthermore, in comparison to our synthetic experiments, the difference between V-UCB Unif and V-UCB Prob is not significant in the robot pushing experiment. This is because the chance that a uniform sample of LV has a large probability mass is higher in the robot pushing experiment due to a larger region of $\mathcal{D}_{\mathbf{z}}$ having high probabilities.

5. Conclusion

To tackle the BO of VAR problem, we construct a no-regret algorithm, namely VAR *upper confidence bound* (V-UCB), through the design of a confidence bound of VAR and a set of *lacing values* (LV) that is guaranteed to exist. Besides, we introduce a heuristic to select an LV that improves the empirical performance of V-UCB over random selection of LV. We also draw an elegant connection between BO of VAR and adversarially robust BO in terms of both problem formulation and solutions. Lastly, we provide practical techniques for implementing VAR with continuous $\mathbf{Z}$. While V-UCB is more computationally efficient than the state-of-the-art ρKG^{apx} algorithm for BO of VAR, it also demonstrates competitive empirical performances in our experiments. For future works, it is of interest to generalize this work to the optimization of CVAR of a black-box function (Cakmak et al., 2020), multi-fidelity BO (Zhang et al., 2017; 2019), high-dimensional BO (Hoang et al., 2018), batch BO (Daxberger & Low, 2017), private outsourced BO (Kharkovskii et al., 2020), and ranking BO (Nguyen et al., 2021).

Acknowledgements

This research is supported by A*STAR under its RIE2020 Advanced Manufacturing and Engineering (AME) Programmatic Funds (Award A20H6b0151).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

Basel Committee on Banking Supervision. International convergence of capital measurement and capital standards, 2006. URL http://www.bis.org/publ/bcbs22.pdf.

Bogunovic, I., Scarlett, J., Krause, A., and Cevher, V. Truncated variance reduction: A unified approach to bayesian optimization and level-set estimation. In *Proc. NIPS*, pp. 1507–1515, 2016.

Bogunovic, I., Scarlett, J., Jegelka, S., and Cevher, V. Adversarially robust optimization with Gaussian processes. In *Proc. NeurIPS*, pp. 5760–5770, 2018.

Cakmak, S., Astudillo, R., Frazier, P. I., and Zhou, E. Bayesian optimization of risk measures. In *Proc. NeurIPS*, 2020.

Chowdhury, S. R. and Gopalan, A. On kernelized multi-armed bandits. In *Proc. ICML*, pp. 844–853, 2017.

Daxberger, E. A. and Low, B. K. H. Distributed batch Gaussian process optimization. In *Proc. ICML*, pp. 951–960, 2017.

Goh, J. W., Lim, K. G., Sim, M., and Zhang, W. Portfolio value-at-risk optimization for asymmetrically distributed asset returns. *European Journal of Operational Research*, 221(2):397–406, 2012.

Hoang, T. N., Hoang, Q. M., Ouyang, R., and Low, B. K. H. Decentralized high-dimensional Bayesian optimization with factor graphs. In *Proc. AAAI*, pp. 3231–3238, 2018.

Iwazaki, S., Inatsu, Y., and Takeuchi, I. Mean-variance analysis in Bayesian optimization under uncertainty. *arXiv preprint arXiv:2009.08166*, 2020.

Kharkovskii, D., Dai, Z., and Low, B. K. H. Private outsourced Bayesian optimization. In *Proc. ICML*, 2020.

Kirschner, J., Bogunovic, I., Jegelka, S., and Krause, A. Distributionally robust Bayesian optimization. In *Proc. AISTATS*, 2020.

Koenker, R. and Bassett, G. Regression quantiles. *Econometrica: Journal of the econometric society*, pp. 33–50, 1978.

Nguyen, Q. P., Tay, S., Low, B. K. H., and Jaillet, P. Top-k ranking Bayesian optimization. In *Proc. AAAI*, pp. 9135–9143, 2021.

Nguyen, T. T., Gupta, S., Ha, H., Rana, S., and Venkatesh, S. Distributionally robust Bayesian quadrature optimization. In *Proc. AISTATS*, 2020.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian processes for machine learning*. MIT Press, 2006.

Scott, W., Frazier, P., and Powell, W. The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM journal on optimization*, 21(3):996–1026, 2011.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proc. ICML*, pp. 1015–1022, 2010.

Torossian, L., Picheny, V., and Durrande, N. Bayesian quantile and expectile optimisation. *arXiv preprint arXiv:2001.04833*, 2020.

Toscano-Palmerin, S. and Frazier, P. I. Bayesian optimization with expensive integrands. *arXiv preprint arXiv:1803.08661*, 2018.

Wang, Z. and Jegelka, S. Max-value entropy search for efficient Bayesian optimization. In *Proc. ICML*, pp. 3627–3635, 2017.

Williams, B. J., Santner, T. J., and Notz, W. I. Sequential design of computer experiments to minimize integrated response functions. *Statistica Sinica*, pp. 1133–1152, 2000.

Zhang, Y., Hoang, T. N., Low, B. K. H., and Kankanhalli, M. Information-based multi-fidelity Bayesian optimization. In *Proc. NeurIPS Workshop on Bayesian Optimization*, 2017.

Zhang, Y., Dai, Z., and Low, B. K. H. Bayesian optimization with binary auxiliary information. In *Proc. UAI*, pp. 1222–1232, 2019.